Interactive Design Coordination for the Building Industry

Technical Memorandum

by

James N. Jackson

June 22, 1970

Civil Engineering Systems Laboratory

Department of Civil Engineering

Massachusetts Institute of Technology

UNCLASSIFIED
Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Project MAC Massachusetts Institute of Technology	2a. REPORT SECURITY CLASSIFICATION UNCLASSIFIED
	2b. GROUP None

3. REPORT TITLE

Interactive Design Coordination for the Building Industry

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

Technical Memorandum

5. AUTHOR(S) (Last name, first name, initial)

Jackson, James N.

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
June 22, 1970	32	14

8a. CONTRACT OR GRANT NO. N00014-69-A-0276-0002	9a. ORIGINATOR'S REPORT NUMBER(S) TM-10
b. PROJECT NO.	
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	

10. AVAILABILITY/LIMITATION NOTICES

Distribution of this document is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	Advanced Research Projects Agency 3D-200 Pentagon Washington, D.C.

13. ABSTRACT

One of the responses to the need for effective interaction in the use of computers for a design project is the supersystem concept proposed for ICES, the Integrated Civil Engineering System. The supersystem is defined as the cooperative effort on the part of the designers of several problem oriented computer capabilities to implement project oriented capabilities by allowing each of their problem oriented subsystems to reference a single file of project data. The supersystem would allow design interaction by having each of the problem oriented computer subsystems reference a single file of information specifying the project.

Future work in the application of computers to interactive and project oriented design in the building industry will have to concentrate on the file structure to be used in the implementation of a computer building design supersystem.

14. KEY WORDS

Computers, Building design, Building construction, ICES System, Structural analysis

DD FORM 1 NOV 65 1473 (M.I.T.)

UNCLASSIFIED
Security Classification

Interactive Design Coordination for the Building Industry

Technical Memorandum

by

James N. Jackson

June 22, 1970

Civil Engineering Systems Laboratory
Department of Civil Engineering
Massachusetts Institute of Technology

Abstract

The problem of effective communication in the process of building design and construction is widely recognized. The involvement of several design disciplines combined with the tendency for designers to work in distinct offices results in little capacity for them to investigate the influence of their design decisions on other design areas.

One of the responses to the need for effective interaction in the use of computers for a design project is the supersystem concept proposed for ICES, the Integrated Civil Engineering System. The supersystem is defined as the cooperative effort on the part of the designers of several problem oriented computer capabilities to implement project oriented capabilities by allowing each of their problem oriented subsystems to reference a single file of project data. The supersystem would allow design interaction by having each of the problem oriented computer subsystems reference a single file of information specifying the project.

Future work in the application of computers to interactive and project oriented design in the building industry will have to concentrate on the file structure to be used in the implementation of a computer building design supersystem.

Acknowledgment

The work described herein was supported by the Office of Naval Research under contract N00014-69-A-0276-0002. The research is being being carried out as a Ph.D. thesis in the Civil Engineering Systems Laboratory of the Department of Civil Engineering, Massachusetts Institute of Technology. The research has been supervised by Professor Kenneth F. Reinschmidt.

The process of building design and construction involves much handling and manipulation of data. What starts out as the single desire of a client for a building develops into a full set of working drawings and specifications by the end of the design phase of the process and ultimately finishes as an existing building. When one examines the data flow in the building process in light of the data manipulating and storage capabilities of the modern electronic digital computer, one expects at first to find a broad utilization of the computer throughout the building industry. Yet when one examines the degree to which computers are actually used in the building process, the findings are generally very disappointing. Few of the design disciplines involved with the building process make any significant use of the computer and even in these few instances, the applications are in completely isolated areas. While many design areas involved in building design have been considered for computer implementation, most efforts have been at the proposal stage only. The two major exceptions have been the areas of structural analysis and construction project scheduling for which large scale systems have been implemented.

The reasons for the pattern of usage that one finds reflect problems both of economics and degree of difficulty. As would be expected, engineers have attacked those problems

first that seemed most promising of solution. Since both structural analysis and construction scheduling are quite straightforward in an analytical sense and require much data processing, they were computerized first. More significant, however, is the fact that these two areas are the exclusive domains of two distinct segments of the building process, the structural engineers and the contractors. Each invested in the software which it felt would make its operations more efficient. Neither was particularly motivated to spend money to make the job of someone else more efficient.

The reasons for this pattern of usage can also be found in the approach taken by designers of computer systems to the whole question of information. The techniques for information handling developed for the analytic problem-solving systems have in the past almost never considered information requirements beyond the scope of the system being implemented. There has been little motivation to consider the information requirements of other systems because, first, there were few enough of these systems implemented on the computer to begin with, and secondly, there had simply been no co-ordination which would result in the information being used even if it were made available. Furthermore, information has generally been structured so as to optimize processing in view of the algorithms used by the subsystem structuring it.

Information has always been considered as a static

collection of data values which were input at the beginning of a computer run and completely purged from the computer at the end of the run. There has been almost no attempt to view information from the point of view of the project, as a highly structured complex which starts as a single idea and which ends after great development as an extremely complex set of drawings and specifications. In those few instances where data has been organized by a computer system on secondary storage, it has been done in such a way as to be of use only to the system which so organized it.

Building data management, then, is an attempt to solve the very complex problems of automating the flow of information between various problem oriented computer capabilities used in the design and construction of buildings, computer capabilities both existing and proposed. Building data management is the concept of data transfer applied to the realm of building systems. Data transfer attempts to make it possible for independently conceived and independently executed computer systems to communicate their results with each other.

Data Transfer

The concept of data transfer is not a new one. The designers of ICES, an acronym for the INTEGRATED CIVIL ENGINEERING SYSTEM (1), have attacked the problem of data transfer from the very beginning of their effort. The ICES

system was visualized as a computer oriented system used by a collection of problem oriented subsystems (2). The analogy was made to a wheel in which the system comprised the axle, the various subsystems the spokes, and data transfer was to have been a kind of rim uniting all of the subsystems via communications capabilites (see Figure 1-1).

However, if one examines Ices System Design (3), the guiding philosophy for the ICES system, one discovers that there are two areas of the system that were not generally implemented. They are the relational data structure capabilities (4) and data transfer. For several years much work was put into the implementation of both of these areas. While some results were obtained in the former area (5), no real working system of any capability resulted in the latter.

In the first efforts to implement data transfer, the ICES researchers attacked the general problem of information flow within the computer. The work was motivated by their strong feeling that subsystem designers should be given full freedom for design of in-core data structures most suited to the problem and algorithms with which they were working. Yet, when these independent systems attempted to each solve a different aspect of the same project, the need arose for them to communicate results with each other. The early work resulted in a proposal for a Data Definition Language (6), but most felt that an appropriate solution to the problem

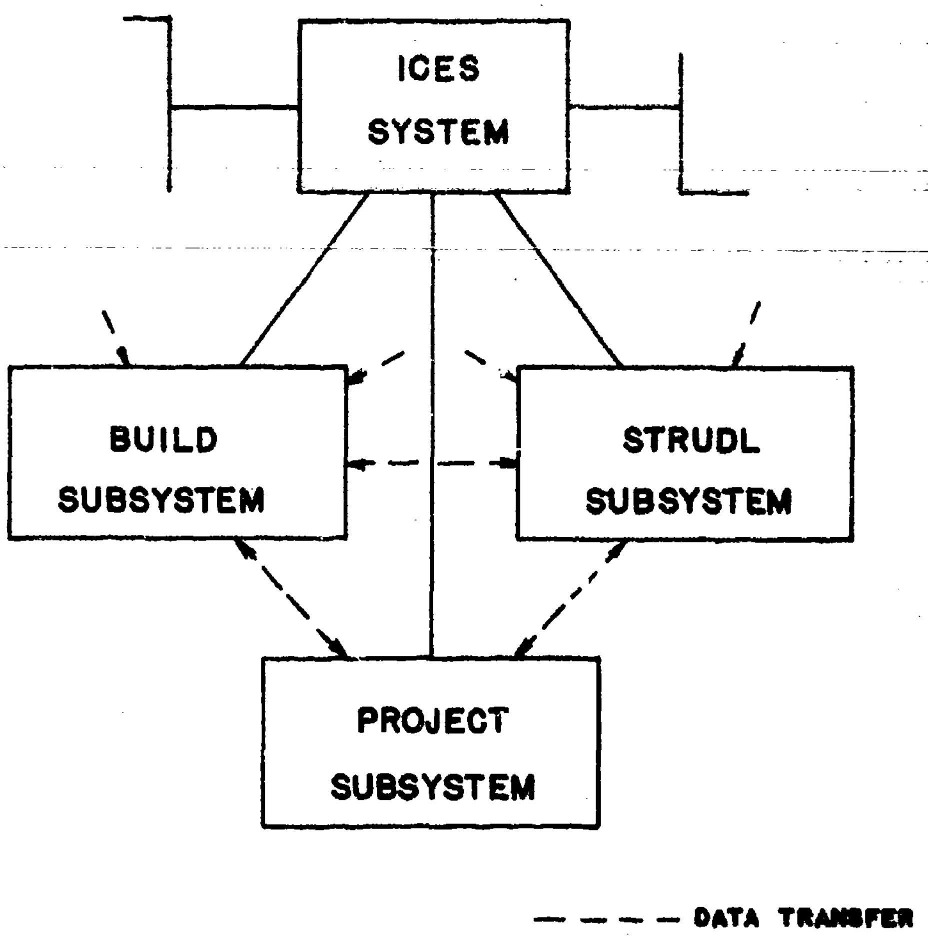


FIGURE I-1
ORIGINAL CONCEPTION
OF
ICES SYSTEM

was still yet to be found. In the interim, of coure, data transfer was actually acomplished by having the engineer using the various subsystems on a project manually transfer information from the printout of one set of results to the problem language input of the second subsystem (See Figure 1-2).

In 1968, Long (7) performed a study of the efforts in data transfer in the context of the ICES system. His major conclusion was that while the attempt to solve the problem of general data sharing between computer systems had borne little fruit, there was some reason to be hopeful that a less general approach to the problem might give better results. He distinguished between the concepts of the system and the subsystem and introduced the concept of a supersystem. The system is comprised of those capabilities, generally oriented toward strictly computer tasks, that are used by all of the subsystems. Subsystems are comprised of capabilities oriented toward some specific engineering problem area. The supersystem is defined as a group of loosely organized subsystems, each oriented toward a specific problem area, but jointly working toward the goal of a project implementation, principally by sharing a common data base stored permanently on a secondary storage device. It is the matter of the orientation, problem versus project, that distinguishes a subsystem from a supersystem. Thus, while STRUDL, the structural design language, is capable of

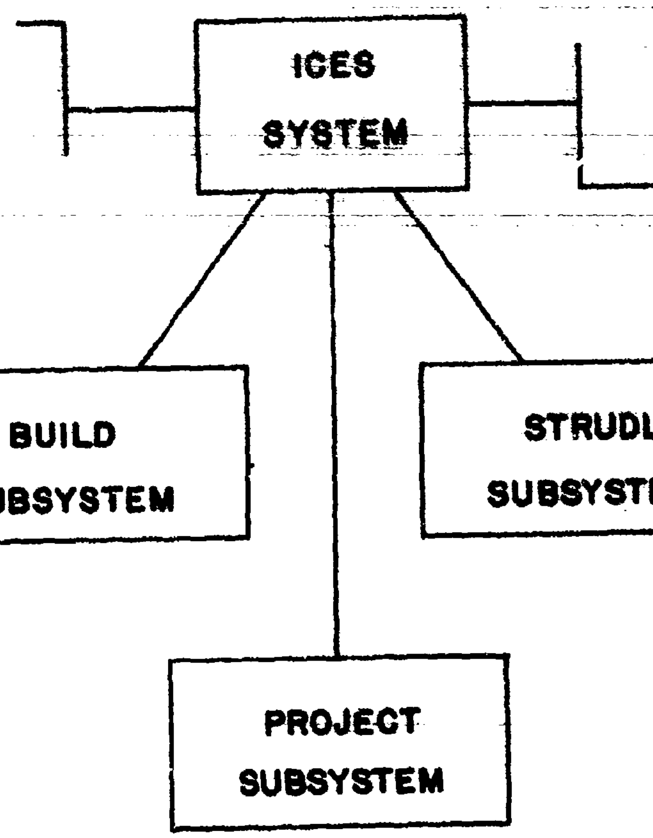


FIGURE I-2
WORKING VERSION
OF THE
ICES SYSTEM

analyzing and selecting members for the structural frame for a building, it is not capable of taking the entire building project or even the structural part from inception to completion. The building design comprises many problem areas, each of which might require a subsystem of the size and complexity of STRUDL.

The implementation of data transfer is important not only for the concept of a supersystem but for the way that engineering is practiced. Engineers, while in school, solve problems. Each problem is a close look at some small, specific engineering task. When the problem is solved, the answer is graded and no more is done with it. Engineers, as practicing professionals, work on projects. They, too, solve problems. In distinction to the work of students, however, the answers to their problems are integrated into the larger project effort. These answers are considered in their ramifications with other "answers" for other problem areas of the project and must be considered as part of all the project data.

Computer efforts in engineering to date have been aimed at giving problem solving capabilities. And just as looking at an engineering project as a series of problems fragments the concept of a project effort, so have these computer capabilities tended to fragment the work that can be done for a project with a computer. This can be observed in the tendency of engineers to require that a problem be of

sufficient size or complexity in order to justify solving it with a computer. The fragmentation has put an artificial barrier between the engineer and his problem solving tool.

Now in order to overcome the tendency toward fragmentation, in order to develop project oriented computer capabilities or supersystems, the whole approach of engineering computer development must be re-examined. Engineering computer technologists can re-orient their efforts and work toward the development of project oriented subsystems - unique, all-encompassing computer systems. These would be large scale efforts and might well result, for example, in a STRUDL-like subsystem for bridge design, another STRUDL-like subsystem for building design, a third for tranmission tower system design, and so forth. The difficulty with this approach is the duplication of effort that is required to develop unique subsystems for each project area. The development of the STRUDL subsystem as a problem oriented capability extended over five years. The duplication of that effort several times for different project areas is worthy of little consideration.

Another approach to implementing project oriented capabilities is specifically that of the supersystem. Each project area would have not a unique computer capability but rather a unique project data structure. Thus computer subsystem developers would continue their current orientation of developing problem solving capabilities. But

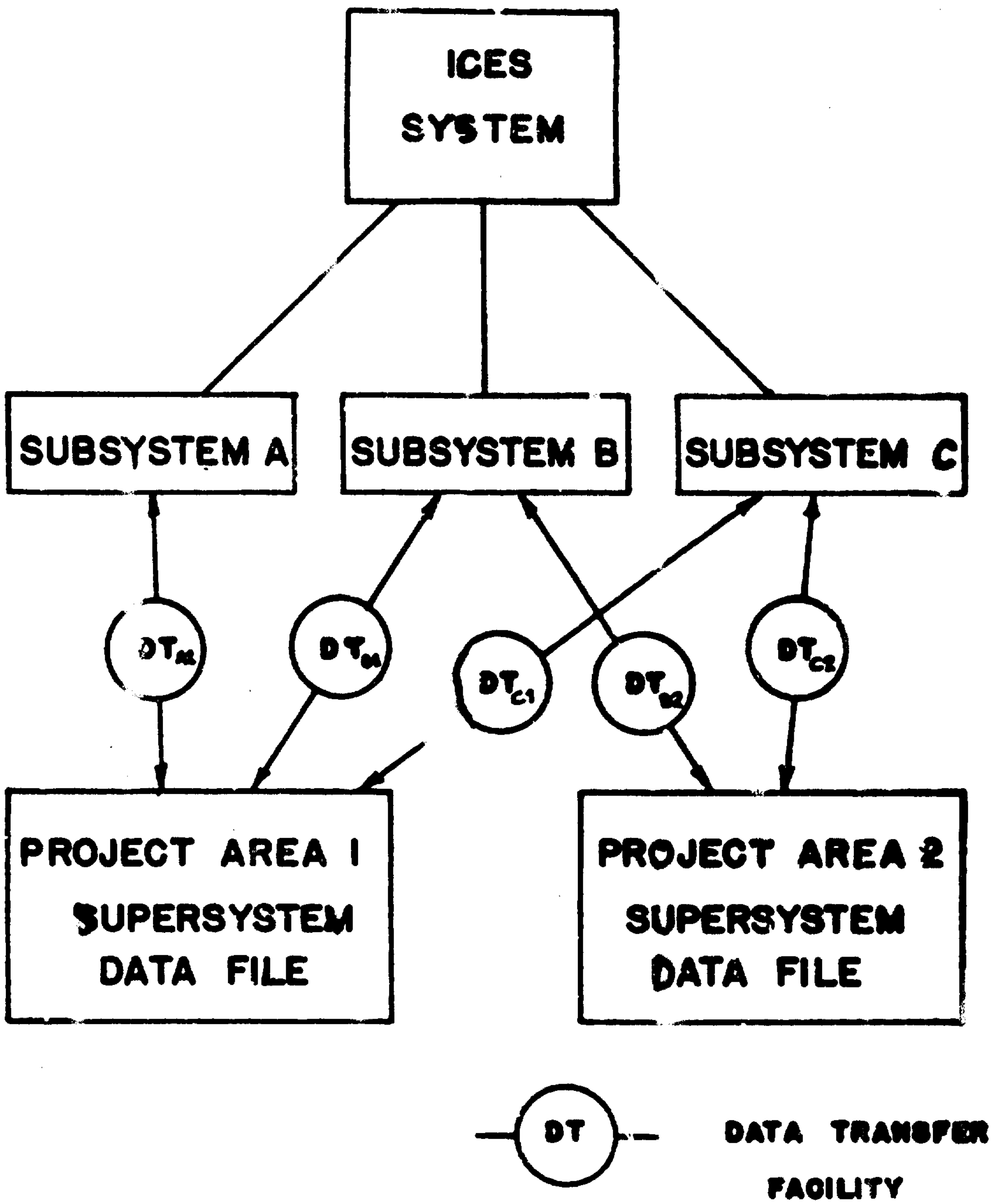


FIGURE I-3

ICES SUPERSYSTEMS

each of these problem solving capabilities would have additional, satellite features that would allow for the implementation of data transfer between the subsystem and a specific project data file. The existing subsystem would be integrated with a new supersystem by the implementation of the satellite data transfer capabilities for the new project data file (See Figure 1-3).

The Building Industry

In the United States, the estimate for the total value of construction during the year 1970 is set at $90 billion (8). Table 1-1 gives a breakdown by project type of the estimated value of building construction during the same year but excluding one and two family dwellings. The same estimate predicts a greater than 100 percent increase in construction value during the decade 1970-1980 to a value of $193 billion. The estimate of the gross national product of the United States for the years 1970 and 1980 given by the estimate are $900 billion and $1,980 billion respectively. Furthermore, in the United States the industry is composed of:

> more than 900,000 contractors and 1,500,000 subcontractors employing over 3,000,000. They are supplied by a myriad of other industries employing large numbers, such as the 240,000 employees of sawmills and planning mills, the 60,000 in millwork and related products, and the 260,000 who manufacture equipment. To handle financial, insurance, and real estate dealings requires another 1,100,000 people of whom more than 600,000 are in real estate alone. The

TABLE I-1 (9)

Forecast of Construction Contracts - 1970

Millions of Dollars

Total Construction *	$52,225
Heavy Construction	$16,875
Non-Residential Building	26,100
Manufacturing	$5,000
Commerical	8,700
Educational	6,000
Medical	2,700
Government Services	1,000
Recreational, Religious, Etc	2,700
Residential *	9,250
Apartments	7,600
Dormitories	900
Hotels and Motels	750

* Excludes one and two families dwellings.

> building design professions include 30,600 registered architects and 75,000 engineers plus a large number of specialists. Manifestly the industry is large but diffuse, and consists of a loose agglomeration of mostly small units. The number of design-construction firms with an annual volume greater than $500 million can be counted on the fingers of one hand. Few materials and equipment producers rank among the nation's 500 largest industrial firms. (10)

The technical areas required in the design and construction of a large building are amazingly diverse. One can consider the professional and economic interests of the building industry as falling into one of four general categories: management, design, construction, and finally operation and maintenance.

The realm of management includes, first of all, the client or owner. The client is the <u>prima movens</u> of the entire industry. It is he who dictates the kind and quality of building depending on his needs and financial backing. Owners range in size from the private, single home builder through developers of capital investment motivated skyscrapers in large metropolitan centers and the Federal government with all of its resources.

Included in the realm of economics, however, are many other professions concerned with building. These include planning boards for urban areas, financiers (including banks, insurance companies, pension and welfare funds, and government mortgage financing agents), real estate developers, zoning commissions, accountants, and the like.

The second realm of the building industry is that of

design. Traditionally, the management of design has been in the hands of an architect who acts as the client's agent for both design and construction. But due to the widely divergent and highly technical nature of many aspects of building design, the architect (excluding one and two dwelling housing, which represents about one-half of the construction dollar value) requires the assistance of professional consultants in the engineering areas. These generally include the structural engineer, the foundation engineer, the mechanical engineer, the electical engineer, and specialists in the areas of cost estimating, interior design, acoustics, illumination, and landscaping.

The third realm of the industry is that of construction. The construction phase of the building project has traditionally been managed by the architect, but the prime agent here is the general contractor. The general contractor, like the architect in the design realm, uses specialized sub-contractors to perform the highly technical phases of the construction. These sub-contractors include plumbers, heating and air conditioning specialists, electrical contractors, plasterers, stone masons, carpenters, roofers, structural steel erectors, and foundation contractors, among others.

The final realm is that of operation and maintenance. Included here are the operations engineers required to keep large mechanical and electrical systems for buildings

functioning properly, cleaning crews, security personnel, and the construction trades required for repairs and modifications.

It should be clear that this diversity of economic interests and intellectual disciplines involved in the building industry lead to a fragmentation that exists on three levels. There is a fragmentation of personnel. The nature of building design alone is such that one can never expect to see a single person being able to do the entire design. There is a fragmentation of location. For the most part as the profession is currently carried on, the participants in the design and construction stages each have separate offices, sometimes even to the extent of being located in different cities. And finally, there is a fragmentation of goals. What may well be the best structural design can lead to a definitely sub-optimal mechanical design, and vice versa. What appears best in terms of initial cost may be very poor when considered in terms of long term costs.

The major consequences of this fragmentation are three. By far the most important and at present the most widely recognized consequence is the communication problem. Communication is a basic aspect of the design and construction of buildings, whether all of the design participants work in a single office or not. The range of design disciplines dictates that professional interaction

take place. The building process typically starts as the desire of a client for a building and is developed through interviews between the client and the designer, through the various design stages, to a fully developed set of contract drawings and specifications. A second consequence of fragmentation and one that follows also from the communication problem is that of sub-optimization of design. A less than perfect communication between the principal designers makes it impossible to estimate how their design decisions affect each other and consequently how such decisions affect overall cost for the client. The problem of optimization in building design is as much a matter of communication as it is of mathematics. And finally, a last consequence of fragmentation is duplication of effort. As currently practiced, the duplicate review of drawings and specifications for cost estimating by architects and bidding by contractors is typical of this duplication of effort.

Consider the kinds of incidents that occur in the current state of building design. The structural and mechanical engineers, having arrived at initial, compatible configurations for the structural frame and duct system, return each to his own office where detail design continues. Later the architect informs the mechanical engineer that certain changes have occurred in the specification of materials for an area, thus changing the heat loads and requiring in turn a larger duct servicing the area. If the

mechanical engineer fails to confer with the structural engineer again, as happens sometimes when the design is rushed, the conflict surfaces only when the contractor discovers that the duct is supposed to go through a structural beam.

One of the reactions to this fragmentation has been the tendency of late to combine in one firm all of the principals involved in the building industry - financier, architect, engineers, and contractor. This reunification at least within the same firm helps alleviate some of the problems resulting from the fragmentation. Many of the goals are thereby consolidated and the problem of communication is generally that much lessened.

The supersystem concept discussed above is another reaction to this problem of fragmentation. The supersystem proposes to consolidate all of the information about a project in a central file of data where it is available to all design participants at the same time. Furthermore, the availability or data to all designers potentially allows for studying the effects of design decisions made in one design area on the other aspects of the overall design. Thus engineers can design in terms of overall project goals rather than the more immediate goals of just their own discipline area. Finally, the development of telecommunications for computers whereby engineers using only low cost terminals in their offices can use the power

of large computers and data files literally across the country, will help in the matter of locational fragmentation where it continues to exist.

The Building Process

Having viewed building construction from the viewpoint of an industry, one can take a slightly different approach and view the same thing from the viewpoint of a process. Considered as a process, building is composed of various phases.

The Royal Institute of British Architects (11) has identified twelve stages of building activity. These stages are only an attempt to give a general classification to the phase of activity most prevalent at the instant, and there is no claim that there are distinctly recognizable points of transition between the stages or that all designers are even in the same stage at the same time. The phases of the building process identified by the Institute are:

Inception - First meetings with client and establishment of design team.

Feasibility - Preparation of first outline from interviews with client and assurance that outline is feasible.

Outline Proposal - Further detailed study of client's requirements, costs of project, and approaches to layout, design, and construction.

Scheme Design - Final development of preliminary design, including full design by architect and

preliminary design by engineering designers.

<u>Detail Design</u> - Final decisions on all design matters.

<u>Production Information</u> - Preparation of final design drawings and specifications.

<u>Bills of Quantities</u> - Preparation of Bills of Quantities for construction bids.

<u>Tender Action</u> - Bidding by general contractors.

<u>Project Planning</u> - Construction co-ordination between general contractor and his sub-contractors.

<u>Operations on Site</u> - Actual construction.

<u>Completion</u> - Completion of construction.

<u>Feed-back</u> - Analysis of design, construction, and operation of building during its life.

This distinction between various phases of the building process is important. Clearly, the problems and even the nature of communication differ during the various phases of building. At inception, ideas and data are few, highly unorganized, constantly changing, and even geometry, a fundamental aspect of all building data is in a very fluid state. By the start of preliminary design, most of the geometry has firmed up, and the real problems of communication and interaction among designers become the most important aspects of the information. By final design, the sheer volume of information has become its most critical aspect and it is that aspect which extends through the construction phases. It is this evolving characteristic of building information (the same holds for the information for

any engineering project) that makes the subject of project data management such a difficult one. These same changing characteristics will dictate explicit requirements for any attempt at automated data management as will become evident.

Why Use Computers in the Building Process

The very fundamental question of why the computer should be used at all in the building process is one that should be faced. In this age of mass computerization it might seem strange that such a question be phrased, but as the complexity and cost of proposed computer systems grow, more and more are coming to ask just that question.

The computer with its allied software is just another among many potential tools for those engaged in the building process. Because of its tremendous potential for extremely fast calculations and large capacity data manipulation, however, the computer stands as a particularly significant tool in the collection of the building designer and contractor. As Miller has stated it:

> Computers are the key to a systems approach to civil engineering. The nature of contemporary projects is so large, and there are so many complex factors and components - all these different kinds of information must be tied together, and the only way you're going to do it is by computer. I'm talking about using computers as information management devices and not as merely computational tools. Only about 10% of our problems are computational by nature, the other 90% are problems of information storage, control, and manipulation. (12)

There is little question of the computer's capability

to store information. Consider, for example, the IBM System/360, Model 65, computer. Configured with one million bytes of core storage, a model 2301 drum unit, a model 2314 disk stroage unit, and a model 2321 data cell drive, such a system has nearly 500 million characters of on-line storage: one million characters of the storage can be accessed in less than one microsecond; five million characters of storage can be accessed in less than ten milliseconds; over sixty million characters of storage can be accessed in less than one-tenth of a second; and all of the nearly one half billion characters of storage can be accessed in just over one-half second (13). Understandably, no one yet really has any feeling of how many characters of data would be required to completely describe a building design. But there is little doubt that the computer will meet the task, at least as regards capacity. The situation looks even more hopeful with speculation that the next generation of computer hardware will improve the cost/performance ratio of computer systems by a factor of from six to twelve over the last generation of hardware (14).

Context of the Current Effort

The task of developing automated data transfer for the building industry is truly a monumental one. The size and complexity of the industry combined with the range of disciplines that are involved in financing, designing, and

constructing buildings has lead to much hesitancy to even attempt to tackle the problem. Clearly no one effort will be completely successful in such an undertaking and the current work is just the beginning of what will have to be a long process of research and evolution. The current effort has been as much an attempt to further define the problem as it has to develop a working solution. One of the things that has become clear is that the solution will be an evolutionary process rather than a completely developed working capability from the start. In the current effort, also, the concentration has been placed on the communication of data between engineers concerned with the design of buildings, rather than architects or the construction or operation phases of the building process. The reasons for the emphasis on the engineer rather than the architectural aspects of design are twofold. First, the author is an engineer rather than an architect. But more significantly, architecture is an atheoretical discipline. An architect considers himself to be an artist working in a technical industry. The consequence of this is that architects structure and treat data differently from the way engineers do. Hence, while the designers of an architectural computer system might not be completely happy with the file structure of information that will be considered later, their system could still be capable of feeding information regarding geometry and materials to the data base. These two

information areas are key components in many of the engineering design areas.

Future Work

There are two major areas to be investigated in the implementation of an ICES building design subsystem: data management and file structure.

The concept of using data as the integrating bond for a building system leads directly to the fundamental question of data management. The general problem of data management has been the object of much computer research and development over the past decade. The development of the generalized data management systems leads one to consider their value for the problem of data management in the building context, or at least the appropriateness of their approach to a solution for this problem.

The generalized data management systems have addressed themselves directly to the problem of how does one manage the computer environment where there exists a large corpus of data about some loosely structured logical entity (generally a corporate or military operation) which must be developed and used by a group of independent computer systems, none of which is responsible for all of the data and all of which must share the use of the data. This is exactly the problem faced in the building realm.

While the use of the generalized data management

systems in the context of building systems has some drawbacks, the ICES systems as currently implemented does have some data management capabilities. The ICES TABLE-II system has file structure capabilities and storage and retrieval functions.

The TABLE-II file structuring capabilities are particularly appropriate for the problem of storing dynamic information in a file on secondary storage. This system, like the generalized data management systems, stores information in such a manner that its location and characteristics are remembered by the system. Furthermore, data are identified by name in such a way that one need merely provide the system with the name and the system is able to retrieve not only the value but also information as to what the characteristics of the data are (dimensional units and computer characteristics). Conceptually, the information is structured as a four level tree: table, row, column, and list position.

The feature of having available a file system which uniquely identifies and manages data within the system is of primary importance in the area of building systems (as well as many other systems). The problems of managing a growing corpus of information used by completely independent computer systmes demands that one consider only a data management system capable of treating the information as a growing, dynamic entity. The classical approach of file

structure based on locational conventions is clearly out of the question in such a situation. Such an approach always demands a fixed file large enough to hold the largest amount of data one can design for. Furthermore, it is generally impossible to identify the condition where data values are missing - where they have not been stored yet. The integration of various computer systems for different discipline areas requires that information regarding dimensional aspects of the data be maintained as well as the convenience of automatic conversion of dimensions on request.

The TABLE-II system has the additional feature that there are currently available a collection of storage and retrieval functions for passing information in either direction between an engineering program and a secondary storage file. The TABLE-II subsystem is rather unique among the ICES subsystems: it exists on both the engineer-user level as a problem oriented language subsystem and on the programmer-user level as the collection of storage and retrieval functions.

The file structure for a computer based information system must closely reflect the structure of the data _as it is used_ by the designers. The file structure for a building information system must be based on the characteristics of the use of data by the engineers and the architect. Each of these people has a responsibility which is uniquely his own.

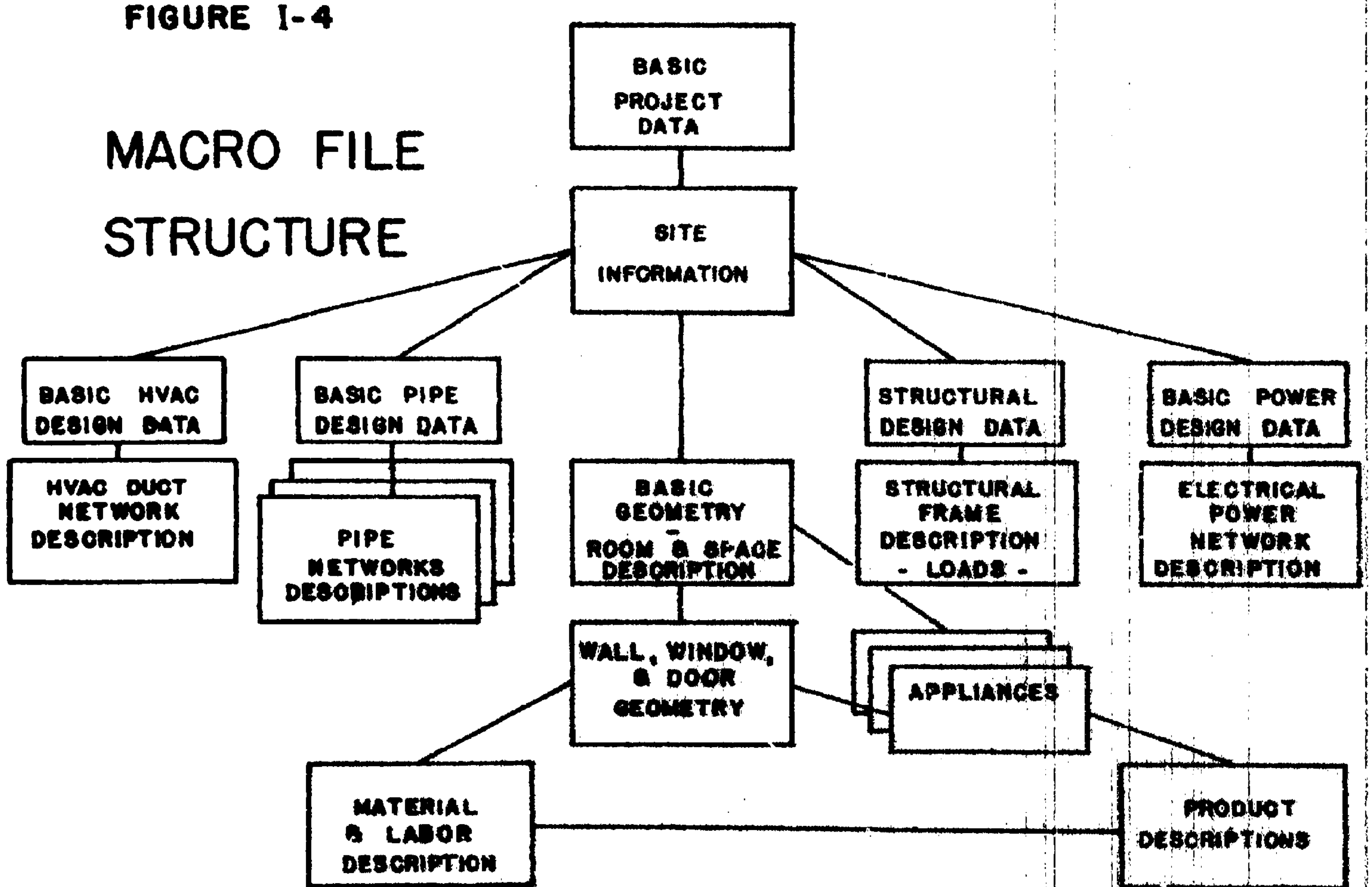
The architect is responsible for the geometry and layout of the spaces as well as the specification of the materials of the walls and other surfaces; the structural engineer is responsible for the structural frame required to support the loads in the building; the electrical engineer for the distribution of electrical power throughout the building as required; the mechanical engineer for the system of air ducts for the delivery of hot and cold air to the spaces and the removal of waste air from the system. Each of these designers has a realm of data which he develops in conjuction with the objectives and requirements of the others connected with the project.

Thus, while the architect generally has the principal concern with windows as an element of form, his decisions on windows greatly influence the heat loads that are the responsibility of the mechanical engineer and the amount of lighting which is the responsibility of the electrical engineer.

The file structure for a computer based information system of building design data must be organized around the use of data by the various agents primarily concerned with that data, and the data within a file should be structured in such a way as to reflect the relational and algorithmic structure of the data. The data regarding windows should be the responsibility of the architect. He is the designer primarily responsible for choosing the quality and location

FIGURE I-4

MACRO FILE STRUCTURE



of windows. Also the location of information about the windows among all of the data items which fall into the realm of the architect should reflect the fact that windows are located in walls, walls which delimit two spaces.

With a file so structured, the mechanical engineer in doing heat load analysis can interrogate the data base of the architect regarding the room under consideration, asking for the U-factors for each of the walls and be told that a particular wall has a window of some specific size and that the design temperature minimum on the other side of that window during the window is -20 degrees Fahrenheit. Furthermore, the electrical engineer can query the same file of the architect and learn that the room has a window with a southern exposure and hence has a calculable flux of sunshine.

The macro level file structure proposed for a building environment is outlined in Figure 1-4. This represents a first effort at a file structure of this sort. As work proceeds, further refinements on the various sub-files and their relationships will become apparent.

Notes

1. See especially, Roos, Daniel, ed., ICES System, General Description, Report R67-49, Department of Civil Engineering (Cambridge: Massachusetts Institute of Technology, 1967), and Jordan, Jane, ed., ICES System, Programmers Reference Manual, Report R67-50, Department of Civil Engineering (Cambridge: Massachusetts Institute of Technology, 1967).

2. Among the subsystems of ICES are several that are of particular concern to the realm of building design and construction. These are the STRUDL subsystem, the PROJECT subsystem, the TABLE subsystem, and the BUILD subsystem. Cf: Logcher, Robert D., et al., ICES STRUDL-II, The Structural Design Language: Engineering User's Manual, First edition, Report R68-91, Department of Civil Engineering (Cambridge: Massachusetts Institute of Technology, 1968); Logcher, Robert D., and James N. Jackson, ICES TABLE-II: An ICES File Storage Subsystem: Engineering User's Manual, First edition, Report R69-34, Department of Civil Engineering (Cambridge: Massachusetts Institute of Technology, 1969); Daniels, Robert L., and E. Jack Hall, ICES PROJECT-II, Project Engineering Control: General Description, Second edition, Report R68-11, Department of Civil Engineering (Cambridge: Massachusetts Institute of Technology, 1968); and Teague, Lavette C., et al., A User's Guide to BUILD, Department of Civil Engineering (Cambridge: Massachusetts Institute of Technology, 1967

3. Roos, Daniel, ICES System Design, Second edition (Cambridge: MIT Press, 1967).

4. A relational data structure, as opposed to a structure based only on data arrays, is one that represents in the computer memory not only data values but also data relationships and structure. This is generally accomplished by the addition of pointer capabilities to the array feature in order to represent the relations.

5. Lipner, S. B., "A List Processing System for Engineering Applications," (M. S. thesis submitted to the Department of Civil Engineering, Massachusetts Institute of Technology, 1966).

6. Roos, ICES System Design, Appendix H.

7. Long, Richard, "Data Transfer in Large Scale Computer Systems," (unpublished Master's Thesis, Sloan School of Management, Massachusetts Institute of Technology, 1968).

8. "Growing Population + Growing Needs = Soaring Construction," Construction Methods, July, 1946, p. 26.

9. Ibid., p. 27.

10. Dietz, A. G. H., The Building Industry, A report prepared for the Commission on Urban Problems, March, 1968. Available from Clearinghouse, PB185208.

11. Ministry of Public Buildings and Works, Professional Collaboration in Designing Buildings, R & D Building Management Handbook, 5 (London: Her Majesty's Stationery Office, 1967).

12. Miller, Charles L., "Decision Makers," interview with Charles L. Miller, Computer Decisions, February, 1970, pp. 12-13.

13. For comparative information on computer capacity and performance, see Auerbach Info., Inc., Auerbach Standard EDP Reports (Philadelphia: Auerbach Info., Inc., 1969).

14. Computerworld, May 20, 1970, p. 1.